

Workflow Management

Christopher
Harrop

Contents

- * What is workflow management?
- * Why do I need workflow management?
- * How does the Workflow Manager at GSD work?
- * Case study

What Is Workflow Management?

- * A workflow is
 - * A collection of tasks that need to get done
 - * A collection of requirements that dictate how those tasks get done
- * A **scientific** workflow is
 - * A collection of scientific computations
 - * The runtime requirements of those computations
- * Workflow management
 - * Automated execution of workflows

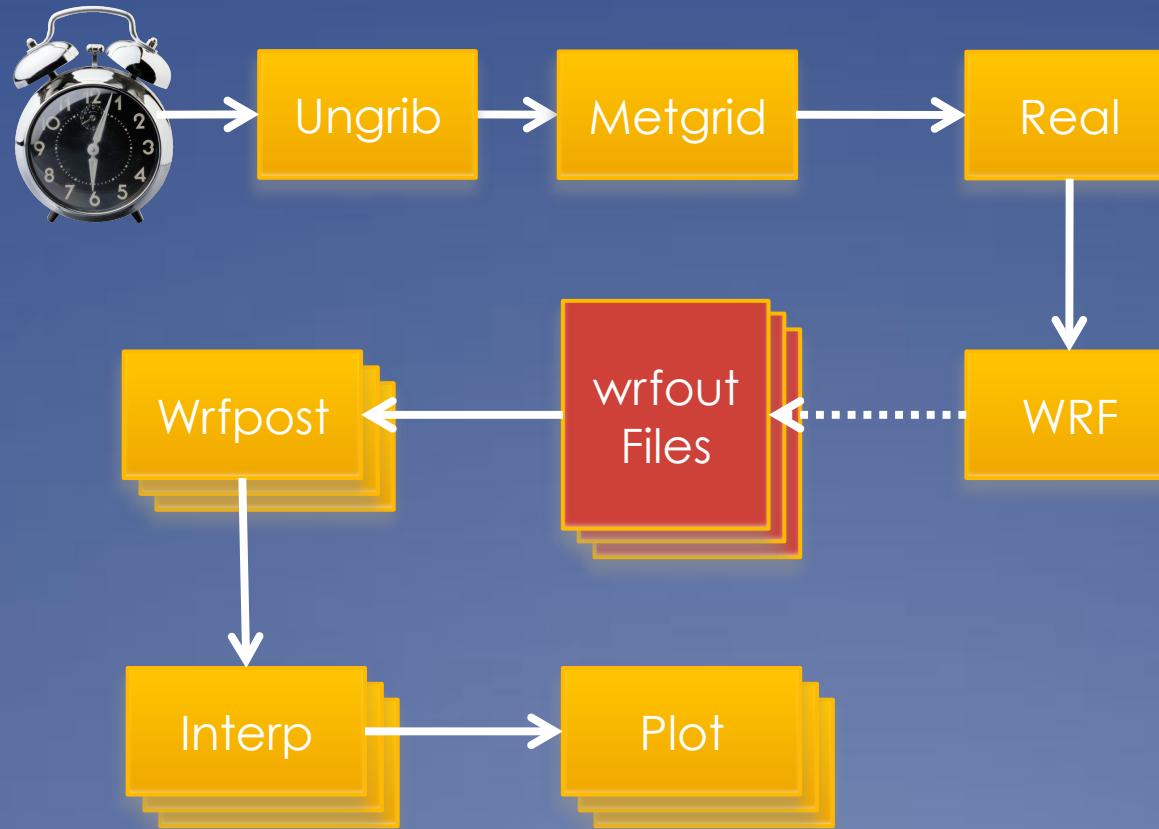
What Is Workflow Management?

- * Not a new concept, used in business since the 1970s
- * Workflow management software capabilities and features vary widely
- * All workflow management software must provide:
 - * A means of describing workflows
 - * Automated execution those workflows

Why You Need Workflow Management

- * End-to-end modeling systems are complicated
 - * Composed of many parts
 - * Complicated interdependencies
 - * Monolithic “mother” scripts are impractical
- * Large scale experiments
 - * 100s to 1000s of end-to-end runs
 - * Each “run” consists of 10s to 100s of jobs
- * Reliability and automation
 - * Automated detection and recovery from failures
 - * Resubmission of failed tasks
 - * Automated tracking of progress of 1000s of jobs

A Conceptual View



Example: 2007 Core Test

# of Tasks	Task Type	Description
5	WPS	3 ungrib tasks, 2 metgrid tasks
4	WRF	2 real tasks, 2 WRF tasks
34	WPP	1 post task per output time per core
32	Bucket	1 Bucket task per output time > 0 per core
34	Interpolation	1 Interpolation task per output time per core
34	Plot	1 plot task per output time per core
17	Difference Plot	1 difference plot task per output time
4	Verification	1 verification task per verification type per core
1	Archival	1 task to archive results
182	Subtotal	Number of tasks for a single end-to-end run
43,680	Total	Number of tasks for entire experiment

GSD's Workflow Manager

- * Comprised of Ruby libraries that “talk” to the batch system
 - * Runs at GSD (jet), TACC (ranger), NCAR (bluefire)
 - * Supports SGE, LSF, and LoadLeveler
- * Users define workflows using a custom XML language
- * Cycle oriented, targeted for weather & climate modeling
- * Reliability is achieved through an iterative approach
 - * Each run of the Workflow Manager is one “iteration”
 - * Each “iteration” has potential to make workflow progress
 - * Total number of “iterations” is nondeterministic
 - * Workflow state is stored on disk

A Mathematical Representation

The WM implements a Finite State Machine

$$\text{FSM} = (Q, \Sigma, \delta, q_o, F)$$

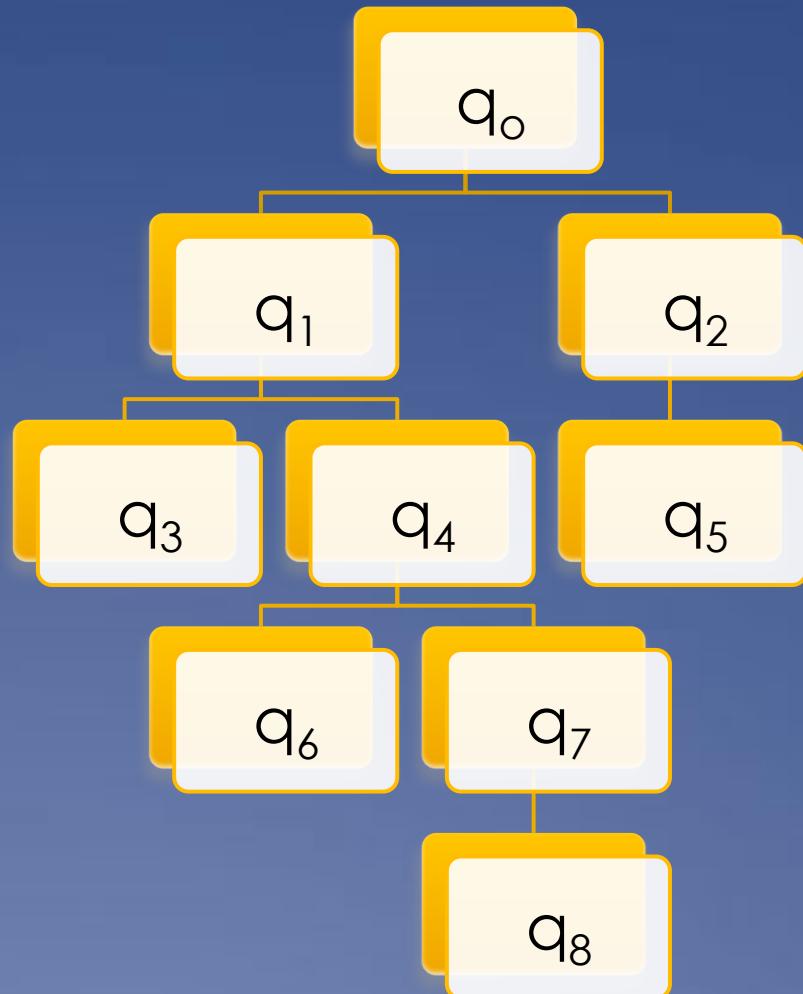
Q = Finite set of states

Σ = Finite set of input events

δ = Function $Q \times \Sigma \rightarrow Q$

$q_o \in Q$ = Initial state

$F \subseteq Q$ = Set of final states



Running the Workflow Manager

- * Test by running on the command line
- * Use cron for production
- * Crontab run times have **NOTHING** to do with when workflow tasks are eligible to run
- * Optimal crontab interval is dependent on task runtimes and product delivery constraints

Running the Workflow Manager

- * Command line usage:

```
workflowmgr.rb --xml=my.xml --store=my.store
```

--xml=The full path of the workflow XML document

--store=The full path of the workflow state file

- * Crontab Usage:

```
*/5 * * * * workflowmgr.rb --xml=my.xml --store=my.store
```

Preview of New Features

- * Compact specification of cycle strings using GNU date flags
 - * E.g. %Y%m%d%H represents cycle in yyyyymmddhh format
- * Use of dd:hh:mm:ss notation in age and offset attributes
- * <Metatask> tags for compact specification of a sequence of nearly identical task definitions

Case Study: Real-time WRF

- * Run every day for 00Z & 12Z cycles
- * Start 00Z cycle at 03:30Z
- * Start 12Z cycle at 15:30Z
- * Post forecasts as the model runs
- * Forecast parameter details left out for clarity



Case Study

```
<?xml version="1.0"?>
<!DOCTYPE workflow
[
]
]>
```

Case Study

```
<?xml version="1.0"?>
<!DOCTYPE workflow
[
]>

<workflow realtime="T">

</workflow>
```

Case Study

```
<?xml version="1.0"?>
<!DOCTYPE workflow
[
]>

<workflow realtime="T">
    <log>/whome/harrop/WRF/log/workflow.log</log>
</workflow>
```

Case Study

```
<?xml version="1.0"?>
<!DOCTYPE workflow
[
<!ENTITY LOG "/whome/harrop/WRF/log">
]>

<workflow realtime="T">

<log>&LOG;/workflow.log</log>

</workflow>
```

Case Study

```
<?xml version="1.0"?>
<!DOCTYPE workflow
[
<!ENTITY LOG "/whome/harrop/WRF/log">
]>

<workflow realtime="T">

<log><cyclestr>&LOG;/workflow_%Y%m%d%H.log</cyclestr></log>

</workflow>
```

Case Study

```
<?xml version="1.0"?>
<!DOCTYPE workflow
[
<!ENTITY LOG "/whome/harrop/WRF/log">
]>

<workflow realtime="T">

<log><cyclestr>&LOG;/workflow_%Y%m%d%H.log</cyclestr></log>
<cycle>2009 * * 0,12 0 0</cycle>

</workflow>
```

Case Study

```
<?xml version="1.0"?>
<!DOCTYPE workflow
[
<!ENTITY LOG "/whome/harrop/WRF/log">
]>

<workflow realtime="T">

<log><cyclestr>&LOG;/workflow_%Y%m%d%H.log</cyclestr></log>
<cycle>2009 * * 0,12 0 0</cycle>

<task id="ungrib" action="ungrib.ksh" tries="3">

</task>

</workflow>
```

Case Study

```
<task id="ungrib" action="ungrib.ksh" tries="3">

<property>
  <name>-A</name>
  <value>rtwrf</value>
</property>

</task>
```

Case Study

```
<task id="ungrib" action="ungrib.ksh" tries="3">

<property>
  <name>-A</name>
  <value>rtwrf</value>
</property>

<property>
  <name>-pe</name>
  <value>serial 1</value>
</property>

</task>
```

Case Study

```
<task id="ungrib" action="ungrib.ksh" tries="3">  
  
  <environment>  
    <name>WRF_ROOT</name>  
    <value>/path/to/wrf</value>  
  </environment>  
  
</task>
```

Case Study

```
<task id="ungrib" action="ungrib.ksh" tries="3">

<environment>
  <name>WRF_ROOT</name>
  <value>/path/to/wrf</value>
</environment>

<environment>
  <name>CYCLE_TIME</name>
  <value><cyclestr>%y%j%H00</cyclestr></value>
</environment>

</task>
```

Case Study

```
<task id="ungrib" action="ungrib.ksh" tries="3">  
  <dependency>  
    </dependency>  
</task>
```

Case Study

```
<task id="ungrib" action="ungrib.ksh" tries="3">  
  <dependency>  
    <timedep><cyclestr>%Y%m%d%H%M%S</cyclestr></timedep>  
  </dependency>  
</task>
```

Case Study

```
<task id="ungrib" action="ungrib.ksh" tries="3">  
  <dependency>  
    <timedep><cyclestr offset="3:30:00">%Y%m%d%H%M%S</cyclestr></timedep>  
  </dependency>  
</task>
```

Case Study

```
<task id="ungrib" action="ungrib.ksh" tries="3">  
  <dependency>  
    <or>  
      <timedep><cyclestr offset="3:30:00">%Y%m%d%H%M%S</cyclestr></timedep>  
    </or>  
  </dependency>  
</task>
```

Case Study

```
<task id="ungrib" action="ungrib.ksh" tries="3">

<dependency>
  <or>
    <timedep><cyclestr offset="3:30:00">%Y%m%d%H%M%S</cyclestr></timedep>
    <and>

      </and>
    </or>
  </dependency>

</task>
```

Case Study

```
<task id="ungrib" action="ungrib.ksh" tries="3">

<dependency>
  <or>
    <timedep><cyclestr offset="3:30:00">%Y%m%d%H%M%S</cyclestr></timedep>
    <and>
      <filedep><cyclestr>%y%h%H00.t%Hz.sanl</cyclestr></filedep>
      <filedep><cyclestr>%y%h%H00.t%Hz.sfcnl</cyclestr></filedep>
    </and>
  </or>
</dependency>

</task>
```

Case Study

```
<task id="ungrib" action="ungrib.ksh" tries="3">

<dependency>
  <or>
    <timedep><cyclestr offset="3:30:00">%Y%m%d%H%M%S</cyclestr></timedep>
    <and>
      <filedep age="120"><cyclestr>%y%h%H00.t%Hz.sanl</cyclestr></filedep>
      <filedep age="120"><cyclestr>%y%h%H00.t%Hz.sfcnl</cyclestr></filedep>
    </and>
  </or>
</dependency>

</task>
```

Case Study

```
<?xml version="1.0"?>
<!DOCTYPE workflow
[
<!ENTITY LOG "/whome/harrop/WRF/log">
]>

<workflow realtime="T">

<log><cyclestr>&LOG;/workflow_%Y%m%d%H.log</cyclestr></log>
<cycle>2009 * * 0,12 0 0</cycle>

<task id="ungrib" action="ungrib.ksh" tries="3">

<property><name>-A</name><value>rtwrf</value></property>
<property><name>-pe</name><value>serial 1</value></property>

<environment><name>WRF_ROOT</name><value>/path/to/wrf</value></environment>
<environment><name>CYCLE_TIME</name><value><cyclestr>%y%j%H00</cyclestr></value></environment>

<dependency>
  <or>
    <timedep><cyclestr offset="3:30:00">%Y%m%d%H%M%S</cyclestr></timedep>
    <and>
      <filedep age="120"><cyclestr>%y%h%H00.t%Hz.sanl</cyclestr></filedep>
      <filedep age="120"><cyclestr>%y%h%H00.t%Hz.sfcnl</cyclestr></filedep>
    </and>
  </or>
</dependency>

</task>

</workflow>
```

Case Study

```
<task id="metgrid" action="metgrid.ksh" tries="3">  
    <!-- This is a comment.  Properties go here -->  
  
    <!-- This is a comment.  Environment vars go here -->  
  
    <dependency>  
        <taskdep task="ungrib"/>  
    </dependency>  
  
</task>
```

Case Study

```
<task id="real" action="real.ksh" tries="3">  
    <!-- This is a comment.  Properties go here -->  
  
    <!-- This is a comment.  Environment vars go here -->  
  
<dependency>  
    <taskdep task="metgrid"/>  
</dependency>  
  
</task>
```

Case Study

```
<task id="wrf" action="wrf.ksh" tries="3">  
    <!-- This is a comment.  Properties go here -->  
  
    <hangdependency>  
        <filedep age="15:00">rsl.error.0000</filedep>  
    </hangdependency>  
  
    <dependency>  
        <taskdep task="real"/>  
    </dependency>  
  
</task>
```

Case Study

```
<task id="wrf" action="wrf.ksh" tries="3">  
    <!-- This is a comment. Properties and Environments go here -->  
  
    <hangdependency>  
        <filedep age="15:00">rsl.error.0000</filedep>  
    </hangdependency>  
  
    <deadlinedependency>  
        <timedep><cyclestr offset="6:00:00">%Y%m%d%H%M%S</cyclestr></timedep>  
    </deadlinedependency>  
  
    <dependency>  
        <taskdep task="real"/>  
    </dependency>  
  
</task>
```

Case Study

```
<task id="wrfpost_00" action="wrfpost.ksh" tries="3">  
    <!-- This is a comment.  Properties and Environments go here -->  
  
<dependency>  
    <filedep age="120">  
        <cyclestr offset="0">wrfout_d01_%Y-%m-%d_%H:00:00</cyclestr>  
    </filedep>  
</dependency>  
  
</task>
```

Case Study

```
<task id="wrfpost_01" action="wrfpost.ksh" tries="3">  
    <!-- This is a comment.  Properties and Environments go here -->  
  
<dependency>  
    <filedep age="120">  
        <cyclestr offset="1:00:00">wrfout_d01_%Y-%m-%d_%H:00:00</cyclestr>  
    </filedep>  
</dependency>  
  
</task>
```

Case Study

```
<task id="wrfpost_24" action="wrfpost.ksh" tries="3">  
    <!-- This is a comment.  Properties and Environments go here -->  
  
<dependency>  
    <filedep age="120">  
        <cyclestr offset="1:00:00:00">wrfout_d01_%Y-%m-%d_%H:00:00</cyclestr>  
    </filedep>  
</dependency>  
  
</task>
```

Case Study – Metatasks

Designed and Implemented by Robert Olabode

```
<metatask>

<var id="fcst">00 01 02 03 04 05 06
    07 08 09 10 11 12 13
    14 15 16 17 18 19 20
    21 22 23 24</var>

<task id="wrfpost_#fcst#" action="wrfpost.ksh" tries="3">
    <!-- This is a comment. Properties and Environments go here -->

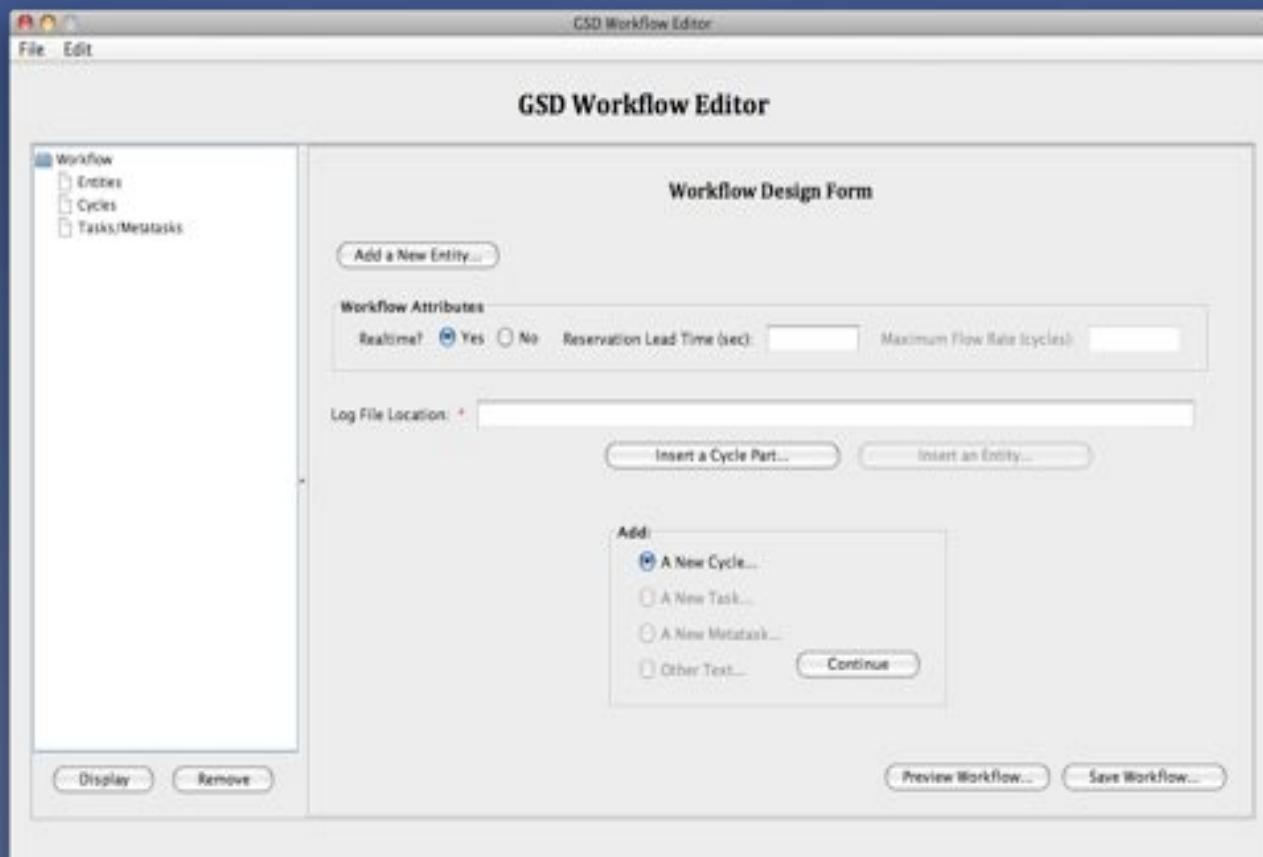
    <dependency>
        <filedep age="120">
            <cyclestr offset="0:#fcst#:00:00">
                wrfout_d01_%Y-%m-%d_%H:00:00
            </cyclestr>
        </filedep>
    </dependency>

</task>

</metatask>
```

GUI Coming Soon

Designed and Implemented by Kawana Fuller



The End?

